REMARKS

Claims 1-31 remain pending. By this Amendment, claims 1, 20, 25-28 and 31 are amended.

Claims 1-19 were rejected under 35 U.S.C. §112, second paragraph. Reconsideration and withdrawal of this rejection are respectfully requested in view of applicants' clarifying amendments to claim 1. The recited sorting of data of interest and storage of that data performs the compacting function referenced in the claim preamble. As amended, the storing of the data of interest is clearly correlated with the determination of whether residual storage space in the first buffer exists.

Claims 1-17, 19-26 and 28 were rejected under 35 U.S.C. §103(a) for alleged obviousness over Kurooka et al. patent application publication No. US2002/0162055 (Serial No. 09/920,930) in view of Litt U.S. Patent No. 6,816,989. Claims 18, 27, 29, 30 and 31 were rejected for alleged obviousness over the asserted combination of Kurooka et al. and Litt, and further in view of Swoboda et al. U.S. Patent No. 6,754,599. These rejections are respectfully traversed.

In accordance with the inventive method for compacting state data of an emulation system of independent claim 1 (and claims 2-19 depending therefrom), a determination is made of whether residual storage space in a first buffer exists, and at least a portion of sorted data of interest from a first sample of state data is stored in the first buffer if it is determined that residual storage space in the first buffer exists. Otherwise, the portion is stored in at least one other buffer. Such a technique is not taught nor suggested by Kurooka et al. or Litt, taken alone or in combination.

Kurooka et al. teach alternating the storage of data into trace buffer memories cyclically in a predetermined order, not based on a determination of whether residual storage space in a first buffer exists. This is seen plainly, e.g., in Fig. 2 of Kurooka et al., wherein eight bit words are sequentially

and alternatingly stored in trace buffer memories A and B without regard to the fullness of those buffers. See also, page 1, paragraph 11 of Kurooka et al., and page 2, paragraphs 29-33.

Litt, on the other hand, determines whether or not to write data over previously stored data in buffers based upon a relative importance of that data. Litt provides no teaching whatsoever of determining if residual storage space in a first buffer exists and storing a portion of data of interest in the first buffer if residual storage space exists, and otherwise storing the portion in at least one other buffer.

Similarly, the apparatus of claim 20 (and claims 21-28 depending therefrom) includes a first select logic device which receives samples of state data, sorts samples of state data, and selects data of interest from the samples. First and second buffers are coupled to the first select logic device and are configured to receive the selected data of interest in an alternating manner by filling one buffer and then the other. Such an apparatus neither taught nor suggested by Kurooka et al. and/or Litt, taken alone or in combination. As mentioned above, Kurooka et al.'s alternating storage of data into trace buffer memories is carried out in a predetermined order as shown, e.g., in Fig. 2, not by filling one buffer and then the other, as recited in claim 20. Likewise, Litt provides no teaching whatsoever of storing selected data alternatively in first and second buffers by filling one, and then the other.

It is noted that the Office Action, while relying on a combination of Kurooka et al. and Litt, does not state the manner in which Litt and Kurooka et al. would be combined to achieve the claimed inventions. Rather, section 5 of the Office Action, at page 5, merely states that "it would have been obvious to one (e.g., a designer) of ordinary skill in the art at the time the invention was made to apply the teachings of Litt to Kurooka." This provides no indication of how or to what end the teachings of Litt would be applied to Kurooka et al. In any event, at best, the combination of Kurooka et al. and Litt would yield a system that alternatively stores data cyclically in a

predetermined order (as taught by Kurooka et al.) and overwrites previously stored data based upon the relative importance of the data (as taught by Litt). Such a combination does **not** yield a method or apparatus as recited in either independent claim 1 or independent claim 20, as described above.

Independent claim 29 is directed to a method for associating trace data chains with pins on an integrated circuit. In accordance with the method, a trace data fill rate of each of a plurality of trace data chains is determined, and a schedule for associating a plurality of pins with the plurality of trace data chains to transfer data out of the trace data chains is determined, based at least upon the determined trace data chain fill rates. As the Office Action acknowledges, Litt does not teach pin scheduling (see discussion of claim 30, Office Action page 12). For this feature, the Office Action turns to Swoboda et al. U.S. Patent No. 6,754,599, and particularly that reference's disclosure of pin manager 261 and pin macros 262 (collectively 260). However, Swoboda et al.'s teachings with respect to pin manager 261 and pin macros 262 does not teach or suggest scheduling an association of plural pins with plural trace chains based on trace data chain fill rates. Swoboda et al. generally disclose that pins may be dynamically allocated between trace and other debug functions. See, e.g., column 16, line 38-column 17, line 42. The brief description of pin manager 261 and pin macros 262 (column 17, lines 44-57) provides no explanation concerning the manner in which pins would be assigned among plural trace chains.

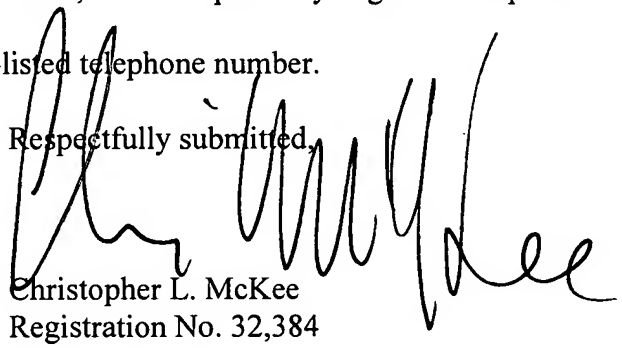
Similarly, the method of independent claim 30 includes the steps of determining fill rates of a plurality of trace data chains based at least upon the determined fill rates, and associating a set of the plurality of trace data chains with the plurality of pins in accordance with the determined schedule. As described above, Swoboda et al.'s teachings with regard to pin manager and pin macros 260 does not teach or suggest determining a schedule for associating pins with plural trace data chains based

upon determined fill rates of the plural trace data chains, instead only generally teaching the dynamic allocation of pins to trace and other debug functionalities.

Independent claim 31 is directed to an apparatus included in an integrated circuit along with an emulator. The apparatus comprises a plurality of trace data chains, a trace pin select logic device, a plurality of pins, and a memory coupled to the trace pin select logic device and configured to store a schedule to associate a selected set of the plurality of trace data chains with the pins based at least upon determined trace data chain fill rates of the set. The asserted combination of Kurooka et al., Litt, and Swoboda et al. lacks at least a memory as recited, which is configured to store a schedule to associate a selected set of trace data chains with the IC pins, based at least upon determined trace data chain fill rates of the set.

For all of the foregoing reasons, it is respectfully submitted that this application is now in condition for allowance. Should the Examiner believe that anything further is desirable in order to place the application in even better form for allowance, he is respectfully urged to telephone applicants' undersigned representative at the below-listed telephone number.

Respectfully submitted,



Christopher L. McKee
Registration No. 32,384

BANNER & WITCOFF, LTD.
1001 G Street, N.W., 11th Floor
Washington, D.C. 20001-4597
Tel: (202) 824-3000
Fax: (202) 824-3001

Dated: April 17, 2006